CSCI 620/Section 01/Mior, Introduction to Big Data, Fall 2201

Assignment 8 – Clustering

Description

In this assignment, we are going to implement the k-means algorithm over MongoDB using IMDB data. We will use the Euclidean distance in which points will be formed by the start year of the movies and their average rating. You should only consider movies (documents whose type='movie'), number of votes are greater than 10,000 for all questions in this assignment and where both startYear and avgRating exist.

Your tasks

1. For each document, create a field called 'kmeansNorm'. This should contain an array in which the first position is the normalized start year and the second position is the normalized average rating. Let v be the value of a field and m and M the minimum and maximum values among all movies. The normalized value of v is computed as (v – m) / (M – m). (Normalization helps adjust for the different scales used in average rating and start year.) **(10 points)**

2. Write a program which selects *k* random documents from genre *g* (*k* and *g* are inputs to your program). The two fields from the previous question should be inserted into new documents in a collection named centroids. Assign the centroid documents IDs from 1 to *k*. You will need to erase any previous documents in this collection if it already exists.

   (Hint: you can use $sample to select random documents.)
   (15 points)

3. Implement one step of the *k*-means algorithm by assigning a new field 'cluster' in each document with the genre *g* (a parameter) _id of the closest centroid. Then update the documents in the centroids collection with the new centroids. **(20 points)**

4. For each of the following genres: Action, Horror, Romance, and Sci-Fi, and Thriller, initialize the cluster centers and run k-means for up to 100 iterations (or until the clusters converge) starting with k=10 up to k=50 with a step of 5 using only movies of that genre. Plot the sum of squared errors vs the value of *k* for each of the genres (i.e. one plot per genre). Provide your code and the plots. **(30 points)**

(continued on next page)

5. For each of the previous genres, decide which one of the computed number of clusters is the best in each case. For each genre, present an example cluster of movies grouped together such that you can explain it.
   (25 points)